

REMOTE DATA PROTOCOL

BACKGROUND OF THE INVENTION

Field of the Invention

5 The invention relates to a host computer system or server, which has a web browser running on it, and the display of the web browser is transmitted via modem and received by a cellular phone connected to a PDA device which displays the image. In particular, the Remote Data Protocol of the invention
10 consists of multiple virtual machines which are contained in the server, and each contains a browser which has applications running in them. Multiple clients are represented by software, which sends the display of each virtual machine to the remote PDA device to be displayed, and also relays information back to the virtual machine from the PDA device.

15 A particular advantage of this method lies in replacing a conventional terminal on a network system with software, which communicates with the PDA device and the server.

Description of the Prior Art

20 Microsoft Remote Data Protocol uses a main server in which a virtual machine runs multiple applications. Each virtual machine is connected to a dedicated terminal or client, which displays the image of the virtual machine. The
25 advantage of this is to avoid a dedicated computer with application programs for each client, where multiple clients may access applications on a server. All data processing is done in the sever and displayed on terminals, which is a cheaper solution for multiple clients using this Remote Data Protocol. The terminals allow
30 keyboard and mouse commands to be transferred to the server on a network system.

SUMMARY OF THE INVENTION

The present invention relates to multiple portable high speed Internet access PDA (Personal Digital Assistant) devices that can access the Internet and World Wide Web as wireless devices, using a RDP (Remote Data Protocol) client and sever system to facilitate multiple PDA users simultaneously on a single server.

A principal embodiment has a Web server connected to the Internet. This server contains a virtual browser that takes the image displayed in the browser and converts this image into a bit map which is compressed, and communicates via telephone lines to a cellular telephone. The cellular telephone is connected to a high speed internet access device commonly referred to as a PDA (Personal Digital Assistant) which is comprised of a display screen, battery and related micro-electronics. This enables the PDA to receive, decompress and view the bit map image sent from the virtual browser, and more importantly, through cellular phone connectivity to be able to input data or commands from the PDA directly onto the server. The host computer or server receives vector information or compressed data in the form of HTML, JPEG, etc., which is displayed on a web page. The virtual browser virtually displays a virtual image on the server. That image, in whole or in parts, is recompressed and sent to the PDA. In particular, the host computer contains an RDP server which has multiple virtual machines contained within, with each virtual machine containing a web browser. Multiple RDP clients interact with the virtual machines with a dedicated virtual machine for each client on the server. Each client is represented by software, which sends the display of the virtual machine to a single remote PDA device via a dedicated modem port. This modem port allows two way communication between a single PDA and a dedicated virtual machine on the server, via a single RDP client. By implementing multiple virtual machines and multiple RDP clients with multiple modem ports, it is now possible to communicate with multiple users of PDA devices on the RDP server to facilitate Internet browsing, electronic message communication, etc. The RDP client relays information received via a modem

port from the PDA, such as mouse clicks or keyboard commands, to the application program in the virtual machine, which is then processed and a refreshed display sent back to the PDA via the same dedicated RDP client and modem port. The browser on a virtual machine relays display information to the dedicated RDP client such as bitmap files, vector files, commands, buffer information, etc. The RDP client then rasterizes some of the information by drawing it into memory and then proceeds to break up this file into smaller blocks of information. These blocks are compressed and sent to the PDA device through a dedicated modem port connected to each RDP client. The PDA would then receive, decompress and assemble the blocks of information in the original order as first received by the RDP client before the RDP client breaks up into smaller blocks. The browser running in each virtual machine rasterizes most of the information which is sent to the RDP client such as text, etc., but the RDP client may have to rasterize other information such as blocks, etc...

The PDA sends specific data to the virtual machine informing of the current location of the displayable area of the PDA screen with respect to the larger image sent from the browser. This is necessary, as the area displayed by the PDA is smaller than the displayable image on the browser window. The RDP client would send compressed blocks of data representing the image to be displayed in order of priority, such that the first blocks sent to and decompressed by the PDA are in the displayable area of the PDA, which is the current area where the user is viewing. The PDA would then decompress blocks surrounding the displayed area in a particular sequence, such as left to right across rows, and store the image in internal memory. The PDA assembles blocks of the image in a virtual page, which comprises the entire image, thus enabling the user to access any part of the present image without communicating constantly with the RDP client. Hence, as the user scrolls across the image in any direction, the blocks of data comprising the image would already be decompressed and assembled priority-wise for instant viewing. Blocks of the image in closer proximity to the displayed area of the PDA screen get decompressed and stored

into internal memory on higher priority than blocks further away. The PDA continuously sends its current location to the virtual machine on the server to keep getting refreshed data instantaneously as the location changes by the user scrolling or sending commands. This enables a rapid refresh rate of the displayed image especially when scrolling, as areas surrounding the displayed image would be decompressed and already stored in memory.

A mouse click or any keyboard command is given priority in communication between the PDA and the RDP server. Such actions from the PDA are sent instantaneously, interrupting the current activities and this action directed to the web browser on the dedicated virtual machine, which sends a refreshed image back to the PDA device. This feature allows the user to have rapid response to commands. If the user initiates a mouse click or keyboard command on the PDA that does not change the displayed image on the RDP server, then the original activities are continued almost instantly with minimum interruption. However, if the image is changed and refreshed on the RDP server, any old image being sent is stopped and the new image is sent immediately to the PDA if it is a full screen image occupying all or part of the PDA display screen. If the refreshed image sent occupies a part of the PDA display screen and also areas outside the display screen, then the portion of the refreshed image that is displayed on the PDA display screen is sent first to the PDA, and blocks of the image are sent to the PDA which are decompressed and stored in internal memory, in order of priority closest to the displayed image, as previously described. Hence the PDA would assemble blocks of the image in the virtual page, which comprises the entire image, starting first with the current location of the PDA display screen then areas around it.

A beacon is sent from the PDA device to the RDP client and vice-versa many times per minute to confirm that a connection is established and maintained, for the duration of use. In the instance a user is disconnected, either the PDA or the RDP client will not receive a beacon and a time-out will be

RDP client would send this block after it has completed sending the current block of information. The RDP client would monitor the acknowledgement of all blocks of information successfully decompressed, including blocks sent again after receiving error messages from the PDA device.

09677857 "100300

0967-1030

Fig. 1 illustrates Prior Art whereby a PDA device is connected to a cellular phone, which communicates wirelessly to a Host Computer.

Fig. 3 illustrates the displayable area of the PDA device with respect to portions of the image, which are sequentially decompressed prior to viewing.

Fig. 5 illustrates a block diagram of the virtual page with respect to the viewing area of the PDA.

Fig. 7 illustrates communication methods between remote devices and host computers.

7

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

To facilitate description, any numeral identifying an element in one figure will represent the same element in any other figure.

- 5 The principal embodiment of the present invention aims to provide a system that allows multiple users operating PDA (Personal Digital Assistant) devices similar to a palm top computer to access the Internet or the World Wide Web (WWW), as demonstrated in **Fig. 4**. It is a further aim of the present invention, to provide a RDP (Remote Data Protocol) client and sever system to facilitate multiple PDA
10 users simultaneously on a single server.

- Prior art is disclosed in **Fig. 1** where Microsoft Remote Data Protocol is demonstrated, using a main RDP server **1** in which virtual machines **2** exist capable of running multiple application programs **3**. Each virtual machine **2** is
15 connected to a dedicated terminal **5** or client on a network system **4**, which displays the image of the virtual machine. The terminals **5** on the network allow input of keyboard **6** and mouse **7** commands to the RDP server **1**, with all data processing done on the server and displayed on the terminals. In this method of prior art, the RDP clients are the terminals on the network. In the present
20 invention however, the RDP clients are represented by software which interact between the virtual machines on the RDP server and the PDA devices which display the image of the RDP clients, in a completely different fashion.

- A general description of the Prior Art is disclosed in **Fig. 2** with further reference
25 to Patent Applications 09/496,172, 09/501,585, 09/504, 809, 09/504,808, and 09/504,807. A host computer **8** is depicted which is connected to the Internet, and that host may also be a Web server. Running in the host computer, is a Web server program **9**. When a remote user **10** requests to view a Web page (or electronic message etc.) the Web server software receives HTML, JAVA, or
30 other types of information and transmits this information to another software, the

Browser Translator **11**. This software translates the information, (i.e. the entire image comprising graphics and text) received in the form of HTML, Java, etc. (as information may be gathered from different sources) and translates it to a black and white bit map or raster image. In another embodiment, the software
5 translates the information into a raster or color image. The image contains the information that would normally be displayed on a single Web page. The translation program therefore, also acts as a virtual browser. The cellular telephone **12** of **Fig. 2** is connected to the high speed internet access device **13** of the invention commonly referred to as a PDA (Personal Digital Assistant)
10 which is comprised of a display screen **14**, battery and related micro-electronics. This enables the PDA to receive, decompress and view the bit map image sent from the virtual browser, and more importantly, through cellular phone connectivity to be able to input data from the PDA directly onto the server **8**. In particular, the host computer or server of **Fig. 2** and **Fig. 4** receives vector
15 information or compressed data in the form of HTML, JPEG, etc., which is displayed on a web page. The virtual browser virtually displays a virtual image on the server by rasterizing the image, or decompressing parts of the image and putting it into memory. That image, in whole or parts, is recompressed and sent to the PDA. The recompressed data format sent to the PDA, is not necessarily in
20 the same format as the compressed data format first received by the server. For example, the incoming data from a Web page may be in the form of JPEG which is decompressed and displayed on the virtual browser. This data is recompressed and sent to the PDA but can be in the form of TIFF G4 or other formats, and not necessarily JPEG as initially received.

25 Another embodiment of the invention involves the server receiving vector information such as HTML or text and then rasterizing it to bit map format. It can then be shown in memory through the virtual browser and is recompressed through a "loss less" method and sent to the PDA.

The information is received by the device **13** in **Fig. 2**, which has the ability to display a monochrome or color image **15**, in its display window **14**. The information is decompressed and displayed in the order of priority such that part of the image **17** of **Fig. 3**, which substantially or completely covers the displayable area **14** of the device, is decompressed and displayed first and then sequentially the portions **18**, **19** and **20** of the image are decompressed, and stored in an internal memory of the device to be displayed later when the user scrolls up, down, or sideways to these parts of the image.

A principle embodiment of the present invention of **Fig. 4** contains the host computer **21** which contains an RDP server **22** which is a software unit having multiple virtual machines **23** contained within, with each virtual machine containing a web browser **24**. Multiple RDP clients **25** interact with the virtual machines with a dedicated virtual machine for each RDP client. Each RDP client **25** is represented by software, which sends the display of each virtual machine **23** to a single remote PDA device **26** via a dedicated modem port **27**. Each modem port **27** allows a two-way communication between a single PDA **26**, connected to a cellular phone **64**, and a dedicated virtual machine **23** on the RDP server, via a single dedicated RDP client **25**. By implementing multiple virtual machines and multiple RDP clients with multiple modem ports, it is now possible to communicate with multiple users of PDA devices on the RDP server to facilitate Internet browsing, electronic message communication, etc. The RDP client **25** relays information received from the PDA **26**, such as mouse clicks or keyboard commands, via a modem port **27** to the Browser **24** in the virtual machine **23**, which is then processed and a refreshed display sent back to the PDA via the same dedicated RDP client and modem port. The browser on each virtual machine relays display information to the dedicated RDP client such as bitmap files, vector files, commands, buffer information, etc. The RDP client then rasterizes some of the information by drawing it into memory and then proceeds to break up this file into smaller blocks of information. These blocks are compressed and sent to the PDA device through a dedicated modem port

connected to each RDP client, as further illustrated in **Fig. 5**. The PDA would then receive, decompress and assemble the blocks of information in the original order as first received by the RDP client before the RDP client breaks up into smaller blocks. The browser running in each virtual machine rasterizes most of the information which is sent to the RDP client such as text, etc., but the RDP client may have to rasterize other information such as blocks, etc.

The PDA **26** of **Fig. 4** sends specific data to the virtual machine **23** informing of the current location of the displayable area **14** of the PDA screen with respect to the larger image or virtual page **28** sent from the browser to the RDP client then to the PDA, as illustrated in **Fig. 5**. This is necessary, as the area displayed **14** by the PDA is smaller than the displayable image on the RDP client, referred to as the virtual page **28**. The dedicated RDP client would send compressed blocks of data representing the image to be displayed in order of priority, such that the first blocks sent to and decompressed by the PDA are in the displayable area **14** of the PDA, which is the current area where the user is viewing. The PDA would then decompress blocks surrounding the displayed area **14** in a particular sequence, such as left to right across rows, and store the image in internal memory. The PDA assembles blocks of the image in a virtual page **28** stored in the PDA's internal memory, which comprises the entire image, thus enabling the user to access any part of the present image without communicating constantly with the RDP client. Hence, as the user scrolls across the image in any direction, the blocks of data comprising the image would already be decompressed and assembled priority-wise for instant viewing. Blocks of the image in closer proximity to the displayed area of the PDA screen get decompressed and stored into internal memory on higher priority than blocks further away. To explain in detail by referring further to **Fig. 5**, the image displayed on the PDA screen **14** can be comprised of blocks **48, 49, 50, 53, 54 & 55**, which are sent from the RDP client first to the PDA, decompressed and then stored into memory first on the PDA. Then blocks **42, 43, 44, 45, 47, 52, 57, 58, 59 & 60** would be sent immediately after in that order from the RDP client to the

PDA, to be decompressed and stored in the PDA's internal memory. This enables a rapid refresh rate of the displayed image especially when scrolling, as areas surrounding the displayed image would be decompressed and already stored in memory. The PDA continuously sends its current location to the RDP client to keep getting refreshed data instantaneously, should the location change by the user scrolling around or outside the virtual page 28, to enable the RDP client to always have the viewing area and surrounding blocks sent to the PDA.

A mouse click or any keyboard command on the PDA is given priority in communication between the PDA and the RDP server. Such actions from the PDA are sent instantaneously, interrupting the current activities and this action directed to the web browser on the dedicated virtual machine, which sends a refreshed image back to the PDA device through the RDP client. This feature allows the user to have rapid response to commands. If the user initiates a mouse click or keyboard command on the PDA that does not change the displayed image on the RDP server, then the original activities are continued almost instantly with minimum interruption. However, if the image is changed and refreshed on the RDP server, any old image being sent is stopped and the new image is sent immediately to the PDA if it is a full screen image occupying all or part of the PDA display screen. If the refreshed image sent occupies a part of the PDA display screen and also areas outside the display screen, then the portion of the refreshed image that is displayed on the PDA display screen is sent first to the PDA, and blocks of the image surrounding the PDA displayed image are sent next to the PDA which are decompressed and stored in internal memory, as previously described. Hence the PDA would assemble blocks of the image in the virtual page, which comprises the entire image, starting first with the current location of the PDA display screen then areas around it.

A beacon is sent from the PDA device to the RDP client and vice-versa many times per minute to confirm that a connection is established and maintained, for the duration of use. In the instance a user is disconnected, either

0967857-100300

RDP client informing which block of information needs to be sent again, and the RDP client would send this block after it has completed sending the current block of information. The RDP client would monitor the acknowledgement of all blocks of information successfully decompressed, including blocks sent again after receiving error messages from the PDA device. To illustrate this further, reference is made to **Fig. 6** which shows a string of data blocks **61** to be sent from the RDP client to the PDA device. As the PDA receives the data blocks acknowledgements or error messages **62** are sent back to the RDP client. Since the RDP client responds to all data from the PDA, the string of data blocks actually sent from the RDP client to the PDA is represented by **63**. To further explain the events of the RDP client **63**, blocks **31, 32, 33 & 34** are sent in this order from the RDP client to the PDA, with successful acknowledgements **ack31 & ack32** sent from the PDA to the RDP client, but whilst block **34** is being sent, an error message in block **33** is relayed from the PDA, resulting in the RDP client sending this block **33** again as soon as block **34** is sent. The RDP client resumes sending blocks **35, 36 and 37**, but notices no acknowledgement for block **35** after receiving successful acknowledgements **ack34 & ack36** sent from the PDA, which results in the RDP client sending block **35** after block **37** is sent. The RDP client may also respond faster or slower to acknowledgements or error messages as described above.

The current embodiment as illustrated in **Fig. 7** teaches of a standard serial connection **66** between the PDA device **13** and a cellular phone **12**, with an AT command set for communicating between modems. This allows the cellular phone **12** to act as a modem in communicating with another modem **65** attached to the host computer **8**. Web pages **67** received from the Internet are converted to G4 files **68** then sent via modem to the PDA device.

For a different type of cellular phone that does not allow AT command set communication but provides a TCP/IP Internet connection, another embodiment of the invention is disclosed, as illustrated in **Fig. 8**. The PDA device would

